

## *ObjectStore Version 6.2 Released*

ObjectStore Release 6.2 is now available. Version 6.2 continues the “platform” release of the ObjectStore family of products, including the enterprise-class ObjectStore product, the small footprint product, PSE Pro, and the layered tools, the ObjectStore Active Toolkit, and ObjectStore Inspector.

Additional capabilities have been included in the core ObjectStore product to continue on the theme of performance, scalability and accessibility:

- Support for Java Data Objects (JDO)
- Schema Evolution and Database Compaction
- JMTL Proxy Generator
- Database Verification Optimization
- I/O Stream Library Upgrade

### Support for Java Data Objects (JDO)

Version 6.2 adds support for the Java Data Objects (JDO) 1.0.1 specification to the ObjectStore product. With ObjectStore JDO, developers now have two rich development interfaces, the OSJI and now the JDO interface. Both interfaces enable developers to transparently and transactionally persist data in their Java applications. OSJI provides that transparent persistence while enabling access to the rich capabilities of ObjectStore; JDO provides similar capabilities while enabling application portability through an industry-standard manner.

When coupled with the ObjectStore storage engine, the ObjectStore JDO interface lets you leverage ObjectStore’s patented Cache Forward™ Architecture to create high performance real-time data management solutions via an industry standard interface. The ObjectStore JDO interface has been extended to include multi-database support and to enable MVCC access, to allow for greater performance and concurrency in your Java applications.

You need JDK 1.4 or later to use JDO with ObjectStore. For details, see the *ObjectStore Java API User Guide* available online at [www.objectstore.com/techsupport/documentation/objectstore](http://www.objectstore.com/techsupport/documentation/objectstore).

### Schema Evolution and Database Compaction

Schema evolution and database compaction have been completely reimplemented in ObjectStore 6.2. Schema evolution now takes advantage of modern hardware resources such as multiple CPUs and large main memory, and is aware of the non-uniform memory access speed caused by multiple levels of hardware caching. ObjectStore 6.2 schema evolution has been measured to be upwards of 100 times faster than ObjectStore 6.0 on large databases, yielding evolution times in excess of 7Gb/hr.

In the past, when application changes required schemas to evolve, that evolution had adverse effects on object locality - a critical design and performance element of your application architecture. As part of this release, schema evolution includes a major, and long-requested, enhancement to eliminate that object de-clustering that had severely impacted some customers. With ObjectStore 6.2, object locality is retained, and even though page boundaries may change as a result of object growth, objects continue to remain close together, on adjacent pages if not on the same page.

This work represents the initial phase of a multi-release strategy to overhaul schema evolution and related tools. Some additional features planned for schema evolution that are not included in this initial release are support for pointer-to-member and instance reclassification, support for remote schema, and support for cluster splitting to avoid the 2GB limit.

## JMTL Proxy Generator Tool

New with ObjectStore 6.2 is a tool that automatically wraps your application classes with a Proxy class to add transaction logic for object persistence. This enables you to focus on your application business logic, instead of persistence and transaction management, while gaining the additional performance and scalability offered by the JMTL.

The proxy generator tool works by wrapping your selected classes, those that manipulate persistent objects, with a Proxy class – it does not modify your classes. With this tool, JMTL now provides the ability to support entity bean persistence, and the ability for a session bean to manage its own persistent data. Currently, integration is provided for Weblogic 7.0 and 8.1, and there is the facility to develop integrations with other EJB Servers. Contact ObjectStore technical support for these integrations.

## Database Verification Optimization

In conjunction with the rework of the schema evolution capability, the database verification utility, `osverifydb`, has also been reworked to address performance issues and other customer requests. It introduces a four-phase algorithm that guarantees every object/pointer in the database is validated. Page access has been optimized so that target pages are fetched exactly once during verification. This, and other optimizations have led to a 6-fold or better improvement in the overall database verification performance. This enables you to make database verification a more-integral part of your database integrity and system management strategy.

## I/O Stream Library Upgrade

ObjectStore 6.2 for C++ now uses the standard specification for streams I/O in place of the classic I/O streams specification. This change moves the C++ products in step with current standards and addresses lingering incompatibilities between ObjectStore and other third-party products.

## Platform Compatibility

The Support Matrix, found online on the Technical Support web site [www.objectstore.com/techsupport/support\\_matrix](http://www.objectstore.com/techsupport/support_matrix) contains an up-to-date list of all supported and maintained platforms, both compilers and operating systems. Please refer to the Support Matrix if you are in doubt about whether your compiler or operating system is supported.

## Client, Server and Database Compatibility

As in the past, the most recent ObjectStore server can fulfill requests from older ObjectStore clients. In particular, ObjectStore 6.2 servers can fulfill requests from 6.0.x, 6.1.x, and 6.2 clients. However, ObjectStore 6.0.x and 6.1.x servers cannot fulfill requests from 6.2 clients.

Databases created with ObjectStore 6.0.x or 6.1.x may be accessed by ObjectStore 6.2 without a database upgrade. Additionally, you can convert a 6.0.x or 6.1.x database to be used with the ObjectStore JDO interface. To enable JDO client access, additional schema information must be added to the database by first running a utility supplied with the release. Once converted, 6.0.x and 6.1.x clients can no longer access that database.

## Application Compatibility

To upgrade a 6.0.x or 6.1.x application to run under 6.2, you must re-compile and re-link the application. You must modify the source code of an application built with ObjectStore 6.1.x if it meets any one of the following conditions:

- Your application uses classic IO streams (for example, `iostream.h`). You must change your application to use standard streams (for example, `iostream`).
- Your application uses schema evolution APIs. This API was reworked as part of the changes to schema evolution.

Otherwise, all code that previously compiled with ObjectStore 6.1 will continue to compile with 6.2. Additional source code changes may be needed if you are migrating an ObjectStore 6.0.x application. For more information, see the *ObjectStore Migration Guide* available online at

[www.objectstore.com/techsupport/documentation/objectstore](http://www.objectstore.com/techsupport/documentation/objectstore).



[www.objectstore.com](http://www.objectstore.com)

### Worldwide and North American Headquarters

Progress Software Corporation, 14 Oak Park, Bedford, MA 01730 USA Tel: +1 781 280 4000 Fax: +1 781 280 4095

### EMEA Headquarters

Progress Software Europe B.V., Schorpioenstraat 67, 3067 GG Rotterdam, The Netherlands Tel: +31 10 286 5700 Fax: +31 10 286 5777

### UK Office

Progress Software Limited, 210 Bath Road, Slough, Berkshire England SL1 3XE Tel: +44 1753 216 300 Fax: +44 1753 216 301

### German Office

Progress Software GmbH, Konrad-Adenauer-Str. 13, 50996 Köln, Germany Tel: +49-221-93-57-90 Fax: +49-221-93-57-978

ObjectStore and Cache-Forward are trademarks or registered trademarks of Progress Software Corporation in the U.S. and other countries. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Any other trademarks or service marks contained herein are the property of their respective owners.

© 2004 Progress Software Corporation. All rights reserved.